# 日志配置方式一:

这种方式就是将控制台的消息都打印到日志中.

Logback.xml文件中进行配置.

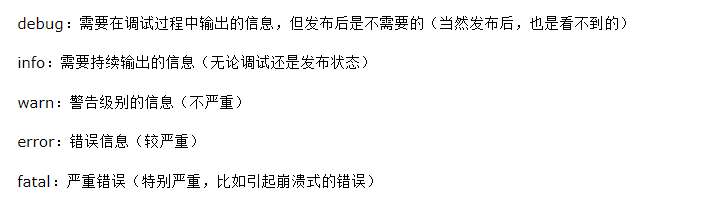
|  |
| --- |
| <?xml version="1.0" encoding="UTF-8"?>  <configuration>  <appender name="STDOUT" class="ch.qos.logback.core.ConsoleAppender">  <encoder>  <pattern>%date{yyyy-MM-dd HH:mm:ss} %highlight(%-5level) (%file:%line\)- %m%n</pattern>  <!-- <charset>UTF-8</charset> -->  </encoder>  </appender>  <appender name="FILE"  class="ch.qos.logback.core.rolling.RollingFileAppender">  <encoder>  <pattern>%date %logger{80}:%line[%level] %msg%n</pattern>  </encoder>  <rollingPolicy class="ch.qos.logback.core.rolling.TimeBasedRollingPolicy">  <fileNamePattern>logs/exchange.%d{yyyy-MM-dd}.log</fileNamePattern>  </rollingPolicy>  </appender>  <root level="INFO">  <appender-ref ref="STDOUT" />  </root>  <logger name="com.thunisoft.jy" level="INFO">  <appender-ref ref="FILE" />  </logger>  </configuration> |

# 日志配置方式二:

这种方式是将不同的日志放在不同的文件中记录下来.

|  |
| --- |
| <?xml version="1.0" encoding="UTF-8"?>  <configuration>  <!-- 设置日志输出路径-绝对路径 -->  <!-- <property name="log.path" value="G:\workspace385\springTest\logs" /> -->    <!-- 设置日志输出路径-相对路径 -->  <property name="log.path" value="logs" />    <!--输出到控制台-->  <appender name="console" class="ch.qos.logback.core.ConsoleAppender">  <encoder>  <pattern>%date{yyyy-MM-dd HH:mm:ss} %highlight(%-5level) (%file:%line\)- %m%n</pattern>  <charset>utf-8</charset>  </encoder>  </appender>    <!--输出到debug-->  <appender name="debug" class="ch.qos.logback.core.rolling.RollingFileAppender">  <rollingPolicy class="ch.qos.logback.core.rolling.TimeBasedRollingPolicy">  <fileNamePattern>${log.path}/logback-debug-%d{yyyy-MM-dd}.log</fileNamePattern>  </rollingPolicy>    <append>true</append>    <encoder>  <pattern>%date{yyyy-MM-dd HH:mm:ss} %highlight(%-5level) (%file:%line\)- %m%n</pattern>  <charset>utf-8</charset>  </encoder>  <filter class="ch.qos.logback.classic.filter.LevelFilter"><!-- 只打印DEBUG日志 -->  <level>DEBUG</level>  <onMatch>ACCEPT</onMatch>  <onMismatch>DENY</onMismatch>  </filter>  </appender>    <!--输出到info-->  <appender name="info" class="ch.qos.logback.core.rolling.RollingFileAppender">  <rollingPolicy class="ch.qos.logback.core.rolling.TimeBasedRollingPolicy">  <fileNamePattern>${log.path}/logback-info-%d{yyyy-MM-dd}.log</fileNamePattern>  </rollingPolicy>    <append>true</append>    <encoder>  <pattern>%date{yyyy-MM-dd HH:mm:ss} %highlight(%-5level) (%file:%line\)- %m%n</pattern>  <charset>utf-8</charset>  </encoder>  <filter class="ch.qos.logback.classic.filter.LevelFilter"><!-- 只打印INFO日志 -->  <level>INFO</level>  <onMatch>ACCEPT</onMatch>  <onMismatch>DENY</onMismatch>  </filter>  </appender>    <!--输出到error-->  <appender name="error" class="ch.qos.logback.core.rolling.RollingFileAppender">  <rollingPolicy class="ch.qos.logback.core.rolling.TimeBasedRollingPolicy">  <fileNamePattern>${log.path}/logback-error-%d{yyyy-MM-dd}.log</fileNamePattern>  </rollingPolicy>    <append>true</append>    <encoder>  <pattern>%date{yyyy-MM-dd HH:mm:ss} %highlight(%-5level) (%file:%line\)- %m%n</pattern>  <charset>utf-8</charset>  </encoder>  <filter class="ch.qos.logback.classic.filter.LevelFilter"><!-- 只打印ERROR日志 -->  <level>ERROR</level>  <onMatch>ACCEPT</onMatch>  <onMismatch>DENY</onMismatch>  </filter>  </appender>    <!--输出到warn-->  <appender name="warn" class="ch.qos.logback.core.rolling.RollingFileAppender">  <rollingPolicy class="ch.qos.logback.core.rolling.TimeBasedRollingPolicy">  <fileNamePattern>${log.path}/logback-warn-%d{yyyy-MM-dd}.log</fileNamePattern>  </rollingPolicy>    <append>true</append>    <encoder>  <pattern>%date{yyyy-MM-dd HH:mm:ss} %highlight(%-5level) (%file:%line\)- %m%n</pattern>  <charset>utf-8</charset>  </encoder>  <filter class="ch.qos.logback.classic.filter.LevelFilter"><!-- 只打印WARN日志 -->  <level>WARN</level>  <onMatch>ACCEPT</onMatch>  <onMismatch>DENY</onMismatch>  </filter>  </appender>    <!--分别设置对应的日志输出节点 -->  <root level="info">  <appender-ref ref="console"/>  </root>    <logger name="com.thunisoft.jy" level="debug">  <appender-ref ref="debug"/>  </logger>  <logger name="com.thunisoft.jy" level="info">  <appender-ref ref="info"/>  </logger>  <logger name="com.thunisoft.jy" level="error">  <appender-ref ref="error"/>  </logger>  <logger name="com.thunisoft.jy" level="warn">  <appender-ref ref="warn"/>  </logger>  </configuration> |

# 级别



# 使用

Logback是与slf4j同时使用的,在java代码中获取logger的方式是在LoggerFactory中获取Logger。

protected static final Logger logger = LoggerFactory.getLogger(TestController.class);

来自于这个包下的：

**import** org.slf4j.Logger;

**import** org.slf4j.LoggerFactory;